


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide



THE ACM DIGITAL LIBRARY

[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

 Terms used **two phase** **commit** and **augment** and **message** and **cache** and **queryable** and **state** and **abort** and **protocol** and **read** **only** and **lock** Fol 21 134,

 Sort results by 
[Save results to a Binder](#)
[Try an Advanced Search](#)

 Display results 
[Search Tips](#)
[Try this search in The ACM Guide](#)
☐ [Open results in a new window](#)

Results 1 - 20 of 20

 Relevance scale ☐ ☐ ☐ ☐

# 1 [Transactional client-server cache consistency: alternatives and performance](#)

Michael J. Franklin, Michael J. Carey, Miron Livny

 September 1997 **ACM Transactions on Database Systems (TODS)**, Volume 22 Issue 3

 Full text available: [pdf\(452.41 KB\)](#)

 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Client-server database systems based on a data shipping model can exploit client memory resources by caching copies of data items across transaction boundaries. Caching reduces the need to obtain data from servers or other sites on the network. In order to ensure that such caching does not result in the violation of transaction semantics, a transactional cache consistency maintenance algorithm is required. Many such algorithms have been proposed in the literature and, as all provide the sam ...

# 2 [Distributed file systems: concepts and examples](#)

Eliezer Levy, Abraham Silberschatz

 December 1990 **ACM Computing Surveys (CSUR)**, Volume 22 Issue 4

 Full text available: [pdf\(5.33 MB\)](#)

 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

The purpose of a distributed file system (DFS) is to allow users of physically distributed computers to share data and storage resources by using a common file system. A typical configuration for a DFS is a collection of workstations and mainframes connected by a local area network (LAN). A DFS is implemented as part of the operating system of each of the connected computers. This paper establishes a viewpoint that emphasizes the dispersed structure and decentralization of both data and con ...

# 3 [Highly available systems for database applications](#)

Won Kim

 March 1984 **ACM Computing Surveys (CSUR)**, Volume 16 Issue 1

 Full text available: [pdf\(2.43 MB\)](#)

 Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#), [review](#)

# 4 [A new approach to developing and implementing eager database replication protocols](#)

Bettina Kemme, Gustavo Alonso

 September 2000 **ACM Transactions on Database Systems (TODS)**, Volume 25 Issue 3

 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Full text available:  [pdf\(449.43 KB\)](#)

[review](#)


Database replication is traditionally seen as a way to increase the availability and performance of distributed databases. Although a large number of protocols providing data consistency and fault-tolerance have been proposed, few of these ideas have ever been used in commercial products due to their complexity and performance implications. Instead, current products allow inconsistencies and often resort to centralized approaches which eliminates some of the advantages of replication. As an ...

**Keywords:** database replication, fault-tolerance, group communication, isolation levels, one-copy-serializability, replica control, total error multicast

##### 5 [Multidatabase systems: Overview of multidatabase transaction management](#)

Yuri Breitbart, Hector Garcia-Molina, Avi Silberschatz

November 1992 **Proceedings of the 1992 conference of the Centre for Advanced Studies on Collaborative research - Volume 2**

Full text available:  [pdf\(3.06 MB\)](#)


Additional Information: [full citation](#), [abstract](#), [references](#)

A multidatabase system (MDBS) is a facility that allows users access to data located in multiple autonomous database management systems (DBMSs). In such a system, *global transactions* are executed under the control of the MDBS. Independently, *local transactions* are executed under the control of the local DBMSs. Each local DBMS integrated by the MDBS may employ a different transaction management scheme. In addition, each local DBMS has complete control over all transactions (global a ...

##### 6 [Overview of multidatabase transaction management](#)

Yuri Breitbart, Hector Garcia-Molina, Avi Silberschatz

October 1992 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 1 Issue 2

Full text available:  [pdf\(3.23 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)


A multidatabase system (MDBS) is a facility that allows users access to data located in multiple autonomous database management systems (DBMSs). In such a system, *global transactions* are executed under the control of the MDBS. Independently, *local transactions* are executed under the control of the local DBMSs. Each local DBMS integrated by the MDBS may employ a different transaction management scheme. In addition, each local DBMS has complete control over all transactions (global a ...

**Keywords:** multidatabase, recovery, reliability, serializability, transaction, two-level serializability

##### 7 [Transactions and synchronization in a distributed operating system](#)

Matthew J. Weinstein, Thomas W. Page, Brian K. Livezey, Gerald J. Popek

December 1985 **ACM SIGOPS Operating Systems Review , Proceedings of the tenth ACM symposium on Operating systems principles**, Volume 19 Issue 5


Full text available:  [pdf\(974.32 KB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

##### 8 [Transaction management issues in a failure-prone multidatabase system environment](#)

Yuri Breitbart, Avi Silberschatz, Glenn R. Thompson

July 1992 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 1 Issue 1

Full text available:  [pdf\(2.12 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#)

This paper is concerned with the problem of integrating a number of existing, off-the-shelf local


database systems into a multidatabase system that maintains consistency in the face of concurrency and failures. The major difficulties in designing such systems stem from the requirements that local transactions be allowed to execute outside the multidatabase system control, and that the various local database systems cannot participate in the execution of a global commit protocol. A scheme based o ...

**Keywords:** algorithms, deadlock recovery, performance, reliability, serializability, transaction log

9 The Recovery Manager of the System R Database Manager

Jim Gray, Paul McJones, Mike Blasgen, Bruce Lindsay, Raymond Lorie, Tom Price, Franco Putzolu, Irving Traiger

June 1981 **ACM Computing Surveys (CSUR)**, Volume 13 Issue 2


Full text available:  [pdf\(1.75 MB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

10 Concurrency Control in Distributed Database Systems

Philip A. Bernstein, Nathan Goodman

June 1981 **ACM Computing Surveys (CSUR)**, Volume 13 Issue 2


Full text available:  [pdf\(3.24 MB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

11 Update propagation strategies to improve freshness in lazy master replicated databases

Esther Pacitti, Eric Simon

February 2000 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 8 Issue 3-4

Full text available:  [pdf\(151.35 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [index terms](#)


Many distributed database applications need to replicate data to improve data availability and query response time. The two-phase commit protocol guarantees mutual consistency of replicated data but does not provide good performance. Lazy replication has been used as an alternative solution in several types of applications such as on-line financial transactions and telecommunication systems. In this case, mutual consistency is relaxed and the concept of freshness is used to measure the deviation ...

**Keywords:** Data replication, Distributed databases, Performance evaluation

12 Reliable transaction management in a multidatabase system

Yuri Breitbart, Avi Silberschatz, Glenn R. Thompson

May 1990 **ACM SIGMOD Record , Proceedings of the 1990 ACM SIGMOD international conference on Management of data**, Volume 19 Issue 2

Full text available:  [pdf\(1.33 MB\)](#)


Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

A model of a multidatabase system is defined in which each local DBMS uses the two-phase locking protocol. Locks are released by a global transaction only after the transaction commits or aborts at each local site. Failures may occur during the processing of transactions. We design a fault tolerant transaction management algorithm and recovery procedures that retain global database consistency. We also show that our algorithms ensure freedom from global deadlocks of any kind.

13 Replication and consistency: being lazy helps sometimes

Yuri Breitbart, Henry F. Korth

May 1997 **Proceedings of the sixteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems**


Full text available:  [pdf\(2.11 MB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

14 Heterogeneous distributed database systems for production use

Gomer Thomas, Glenn R. Thompson, Chin-Wan Chung, Edward Barkmeyer, Fred Carter, Marjorie Templeton, Stephen Fox, Berl Hartman

September 1990 **ACM Computing Surveys (CSUR)**, Volume 22 Issue 3

Full text available:  [pdf\(2.90 MB\)](#)


Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

It is increasingly important for organizations to achieve additional coordination of diverse computerized operations. To do so, it is necessary to have database systems that can operate over a distributed network and can encompass a heterogeneous mix of computers, operating systems, communications links, and local database management systems. This paper outlines approaches to various aspects of heterogeneous distributed data management and describes the characteristics and architectures of ...

15 Object orientation in multidatabase systems

Evaggelia Pitoura, Omran Bukhres, Ahmed Elmagarmid

June 1995 **ACM Computing Surveys (CSUR)**, Volume 27 Issue 2

Full text available:  [pdf\(4.85 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)


A multidatabase system (MDBS) is a confederation of preexisting distributed, heterogeneous, and autonomous database systems. There has been a recent proliferation of research suggesting the application of object-oriented techniques to facilitate the complex task of designing and implementing MDBSs. Although this approach seems promising, the lack of a general framework impedes any further development. The goal of this paper is to provide a concrete analysis and categorization of the various ...

**Keywords:** distributed objects, federated databases, integration, multidatabases, views

16 Implementing atomic actions on decentralized data

David P. Reed

February 1983 **ACM Transactions on Computer Systems (TOCS)**, Volume 1 Issue 1

Full text available:  [pdf\(1.44 MB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

**Keywords:** nested atomic actions, stable storage, time-domain addressing, two-phase commit

17 Overview of an Ada compatible distributed database manager

Arvola Chan, Umeshwar Dayal, Stephen Fox, Nathan Goodman, Daniel R. Ries, Dale Skeen

May 1983 **ACM SIGMOD Record , Proceedings of the 1983 ACM SIGMOD international conference on Management of data**, Volume 13 Issue 4

Full text available:  [pdf\(1.16 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#)


Adaplex is an integrated language for programming database applications. It results from the embedding of the database sublanguage DAPLEX in the general purpose programming language Ada. This paper provides an overview of the DDM: a distributed database manager (DDM) that supports the use of Adaplex as an interface language. The important technical innovations we have incorporated in the design of this system include: 1. An advanced data model that captures

more application semantics than convent ...

**18** A distributed object-oriented database system supporting shared and private databases

Won Kim, Nat Ballou, Jorge F. Garza, Darrell Woelk

January 1991 **ACM Transactions on Information Systems (TOIS)**, Volume 9 Issue 1

Full text available:  [pdf \(1.58 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

ORION-2 is a commercially available, federated, object-oriented database management system designed and implemented at MCC. One major architectural innovation in ORION-2 is the coexistence of a shared database and a number of private databases. The shared database is accessible to all authorized users of the system, while each private database is accessible to only the user who owns it. A distributed database system with a shared database and private databases for individual users is a natu ...

**Keywords:** client-server architecture, federated databases, object-oriented databases

**19** An efficient and reliable reservation algorithm for mobile transactions

Ahmed Elmagarmid, Jin Jing, Omran Bukhres

December 1995 **Proceedings of the fourth international conference on Information and knowledge management**


Full text available:  [pdf \(744.08 KB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

**20** Transactions and consistency in distributed database systems

Irving L. Traiger, Jim Gray, Cesare A. Galtieri, Bruce G. Lindsay

September 1982 **ACM Transactions on Database Systems (TODS)**, Volume 7 Issue 3

Full text available:  [pdf \(1.48 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The concepts of transaction and of data consistency are defined for a distributed system. The cases of partitioned data, where fragments of a file are stored at multiple nodes, and replicated data, where a file is replicated at several nodes, are discussed. It is argued that the distribution and replication of data should be transparent to the programs which use the data. That is, the programming interface should provide location transparency, replica transparency, concurrency transparency, ...




**Keywords:** concurrency control, data partitioning, data replication, recovery

---

Results 1 - 20 of 20

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2004 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)